

White Paper Report

Report ID: 110213

Application Number: HK-50128-13

Project Director: Benjamin Vershbow (benjaminvershbow@nypl.org)

Institution: New York Public Library

Reporting Period: 9/1/2013-8/31/2015

Report Due: 11/30/2015

Date Submitted: 12/3/2015

White Paper Report

National Endowment for the Humanities Project Number: HK-50128-13

Institutions: The New York Public Library and Zooniverse

Program: Digital Humanities Implementation

Project Title: Scribe: An Open Source Framework for Community Transcription

Award Amount: \$325,000

Grant Period: 09/01/2013 – 08/31/2015

Project Director: Ben Vershbow (Director, NYPL Labs) – benjaminvershbow@nypl.org

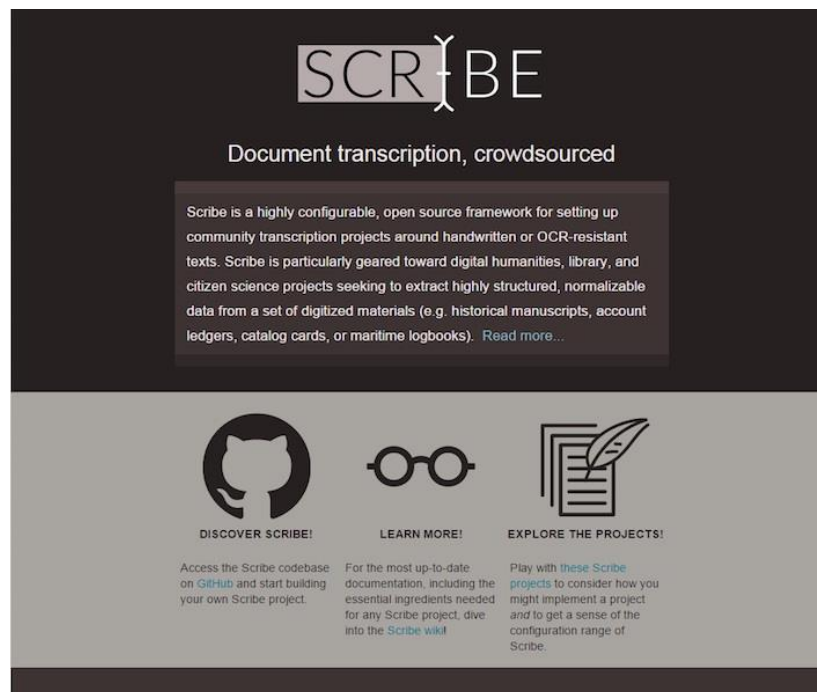
Project Websites:

- <http://scribeproject.github.io/>
- <https://github.com/zooniverse/scribeAPI>
- <http://emigrantcity.nypl.org>
- <http://measuringtheanzacs.org>
- <http://whaling.oldweather.org>

I. Project Summary:

Through the Scribe project, The New York Public Library (NYPL) and Zooniverse set out to create an open source, developer-ready framework for setting up community transcription projects, particularly those aimed at extracting *structured* data from handwritten or OCR-resistant materials. The Scribe code base builds on prior projects and prototypes produced by Zooniverse — a leader in participatory, web-based citizen science that has run over 30 crowdsourcing projects — and NYPL Labs, The New York Public Library’s digitization and library innovation program, which has developed celebrated crowdsourcing initiatives around a wide range of heritage materials including maps, AV collections, historical restaurant menus, and more.

The primary product of this grant is the open-source Scribe code base, which is publically available in a Github repository located at <https://github.com/zooniverse/scribeAPI>. In addition to the code base, there is a 25-page Wiki documenting it, providing step-by-step instructions to developers looking to implement Scribe-powered projects as a way to process and extract data from digitized source materials. The code and documentation are findable via the Scribe landing page, <http://scribeproject.github.io/>, along with links to the three projects NYPL and Zooniverse developed with Scribe during the grant period. These projects— Zooniverse’s *Old Weather: Whaling* (<http://whaling.oldweather.org>) and *Measuring the Anzacs* (<http://measuringtheanzacs.org>), and NYPL’s *Emigrant City* (<http://emigrantcity.nypl.org>)— showcase Scribe’s highly configurable workflows and intuitive user interfaces.



Scribe is not a crowdsourcing project in a box, but it provides the foundation of code for developers to configure and launch a community transcription project far more easily than if starting from scratch. Our hope is that Scribe will significantly lower barriers to entry for digital humanists, scientists, genealogists, journalists, and those working in libraries, archives, and museums to explore crowdsourcing as a means of collection enrichment, data analysis, and resource creation. Institutions or individuals that meet the following three criteria should be able to easily start a Scribe-powered project of their own:

1. Have a collection of digital images that they would like to extract information from, but do not have the resources to do so manually.

2. Are not looking for full text transcription of these images; rather, they would like to collect specific partial text or metadata from the images.
3. A member of their team has basic web development experience, specifically with creating a Rails web application.

Because many interesting and unique documents reside in the collections of small museums and local historical societies, Zooniverse and NYPL Labs anticipate that the availability of the Scribe toolkit could open up myriad user engagement and research possibilities over the long-term. Further use and evaluation of the tools by such institutions will help reveal the next development priorities that will help Scribe to mature into a core part of the digital humanities and citizen science tools ecosystem.

II. Product:

Informed by years of crowdsourcing experience at Zooniverse and NYPL Labs, Scribe proposes a rough grammar for describing materials, workflows, tasks, and consensus. There is significant room for Scribe to evolve further, but we believe that the current iteration represents a strong, first-draft framework for supporting the fundamental work shared by many community transcription projects.

Fundamentally, Scribe is built with the recognition that no two transcription projects are quite alike — that there are nuances of materials (often *within* a single collection of documents) that demand different task flows and prompts — and likewise different aptitudes, appetites, and tolerances in the audiences you may wish to engage as participants in a particular project. With this in mind, Scribe aims to empower developers and project directors to design custom workflows to suit their particular audience and data extraction goals, while the software handles many of the more generic back-end operations and interface templating that is common across projects.

Whereas many transcription tools ask a user to tackle an entire page or document, Scribe breaks the crowd's work into granular tasks. This makes otherwise daunting work more approachable, and allows participants to begin making meaningful contributions almost instantly. Tasks can be woven into three types of workflows: **Marking** (sorting document types and identifying document structures), **Transcription** (data entry), and **Verification** (quality control). This may not be the optimal approach for all transcription projects, but it is particularly well suited to projects in which the target documents are highly structured (e.g. government records, account ledgers, catalog cards, or maritime logbooks), and for which the desired output is a structured data set. Scribe also brings to the table a built-in transcription analysis engine that determines consensus among contributors, further helping to ensure quality data output.

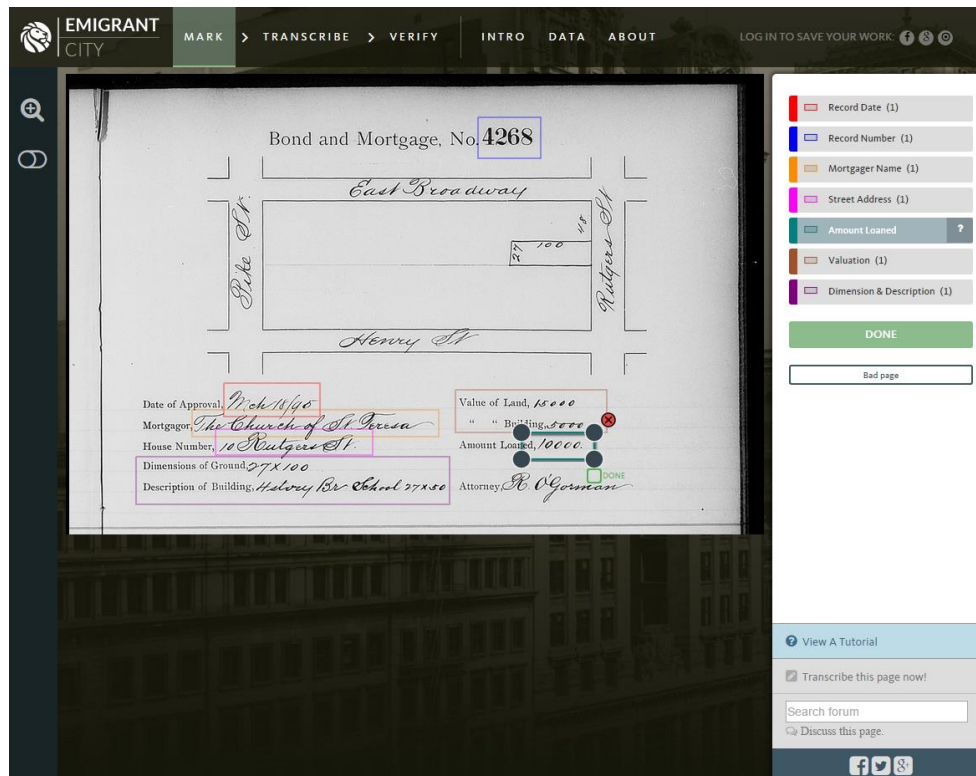
An example of the Scribe workflows applied to a diary page to extract dates might look like:

- A user marks where the date is written in a page of a diary.
- A user transcribes the date that was marked.
- Another user verifies that the date was transcribed accurately.

From the Scribe system's perspective, everything can be reduced to “subjects” and “classifications.” Subjects are the things to be acted upon; classifications are created when you act. Creating a classification has the potential to generate a new subject, which in turn can be classified, which in turn may generate a subject, and so on.

In the case of NYPL's Scribe-powered *Emigrant City* project, individual mortgage bond records are represented as subjects and users are asked to classify, or Mark, specific information fields contained in those records (e.g. record number, mortgager name, property address). When participants mark those

fields, they produce “mark” subjects, which then appear in the Transcribe workflow queue. In the Transcribe workflow, other contributors transcribe (another form of classification) the text they see (highlighted by the previous user’s marking), which are combined with others’ transcriptions as “transcribe” subjects. If there is disagreement among the transcriptions, the transcribe subjects are offered to other users in the Verify workflow, where additional classifications are added by other contributors as votes for the most accurate transcription. This is the configuration that made sense for *Emigrant City*, but Scribe lays the groundwork to support many other configurations.



Emigrant City: marking interface

Not all image collections are ideal for transcription using the current iteration of Scribe. The right type of image collection has the following traits:

- **Unambiguous** - If you generally know what the images contain and what you're looking for, you can craft clear and concise tasks for your users.
- **Consistently formatted** - Consistent image sizes, dimensions, and layouts will enable simpler and more intuitive interfaces for your users.
- **Composite, nonsequential** - The images are related, but users can perform tasks on a single image independently from any others. This allows users to focus on a single subject and task at a time and freely switch between subjects and tasks.
- **Single Subjects** - Each image should contain one subject. This will greatly simplify the interface for your users.
- **High resolution** - This is especially important if legibility of text or details may be an issue.

A good deal of thought and work has gone into Scribe, but it is unquestionably still a product in development. We hope you will work with us to improve it. Or fork it and take it in new directions.

III. Process:

Project activities during the grant period (09/01/2013 – 08/31/2015) unfolded roughly as follows:

- Sprint 1: September 2013 - April 2014
- Pause: May 2014 - October 2014
- Sprint 2: November 2014 - March 2015
- Sprint 3: April 2015 - November 2015

Sprint 1: Scribe Concept Solidifies Through Work on Ensemble, Panoptes, and Readymade (September 2013 - April 2014)

When the project period began, Zooniverse was focused primarily on development of a separate crowdsourcing platform and project-building framework called Panoptes. This work produced elements that were shared with the NYPL Labs team, most notably reusable template system Readymade, which inspired the configurable designs and tools that the current iteration of Scribe uses. The Panoptes API incorporates many of the user-configurable, modular, reusable components of Readymade to create a simplified front-end platform which was used to build Scribe.

Meanwhile, NYPL Labs focused this time primarily on developing a new version of *Ensemble* (<http://ensemble.nypl.org>), a transcription project aimed at extracting structured data about historical performances (theater locations, run dates, performers, production roles, etc.) from digitized theater playbills. After the Scribe project kickoff, the Labs team set about building a new version based on initial discussions with Zooniverse about workflow atomization and consensus generation (quality control of transcriptions).

The screenshot displays the Ensemble V.1 web interface. At the top, there's a navigation bar with 'HOME', 'BROWSE PROGRAMS', and 'ABOUT'. Below this, a header section features a large image of a theater playbill. A modal window is open in the center, titled 'Production Staff'. It contains a form with fields for 'role' (set to 'Director') and 'name' (set to 'Ned Wayburn'). To the left of the form are tabs for 'Show Info', 'Person', and 'Other'. To the right, there's a 'Hint' section and 'Examples' of production staff roles. At the bottom of the modal, there's a 'Delete this transcription' link, a checkbox for 'Go to next line after add', and a 'SAVE' button. The background shows a list of programs with titles like 'The Hen Pecks' and 'A Musical Panorama in Six'.

Ensemble V.1 interface: the user is tasked with transcribing the playbill in its entirety into a pre-defined schema

Ensemble had always been a focal use case for NYPL, and this prototyping period proved highly influential on the conceptual development of Scribe, reflecting our evolving thinking about community transcription in general. *Ensemble V.1* was a fork of a now-old Zooniverse precursor codebase, also called

Scribe (<https://github.com/zooniverse/scribe>), with a similar technology stack, but a very different workflow design philosophy. This had been abstracted from *Old Weather* (<http://oldweather.org>), a transcription project focused on extracting historical climate data from old maritime logbooks. Scribe circa 2012 was “schema-driven” whereas Scribe circa 2015 is “question-driven.”

The screenshot displays the 'oldWeather' web interface. At the top, it identifies the project as 'Old Weather is a ZOONIVERSE project' and compares it to 'THE MILKYWAY PROJECT'. Navigation links include HOME, VESSELS, TUTORIAL, TRANSCRIBE, ABOUT, DISCUSS, and a user profile for 'riordan'. A user profile for 'Cadet riordan' shows 0 weather reports on 1 page and 30 weather reports more for promotion to Lieutenant. A 'Concord' vessel profile is also visible, noting its active regions as West Indies, South America, and Alaskan Waters. The central focus is a 'Weather Observation' form with tabs for Date, Location, Weather Observation (selected), Animals, Refueling, Mentions, Sea Ice, and Events. The form contains input fields for Hour (1), Wind Dir (NE), Force (7), Bar Height (50.73), Ther Attached (70), Dry (70), Wet (67), Water, Weather Code, Cloud Code, and Clear Sky, with an 'OK' button. Below the form is a detailed historical logbook table with columns for Date, Time, Reading of Patent Log, COURSES STEERED by Standard Compass, WIND (Direction by Standard Compass, Force, Head, Low-way), BAROMETER (Height in inches, Ther. alt.), TEMPERATURE (Air, Dry, Wet, Surface, Water), State of the Weather by symbols, Forms of Clouds by symbols, Prop of Clear Sky in fathoms, and State of the Sea. The table contains handwritten entries, including 'A. M.', '1', '6.44', 'N.E.', 'N.E.', '7', '50.73', '70', '67', 'No!', 'Clear', and '8'. A Google Map Data Terms of Use link is visible at the bottom right.

Old Weather interface, which produced the precursor Scribe (2012) codebase

Scribe 2012 asked administrators to define their project based on the schema of the data they hoped to produce. Each document was assumed to have a single, flat list of “entities,” each of which had one or more “fields”. The system was flexible enough provided one’s collection was uniform and the schema not too complicated. Scribe 2015 allows one to specify the schema implicitly by designing a series of branching and looping workflow tasks. Thus, while also supporting the functionality of Scribe 2012, Scribe 2015 provides additional flexibility in the schema one declares.

Zooniverse's *Measuring the Anzacs* project, built on the new Scribe (2015), provides a concrete example of these changes in action. In *Anzacs*, volunteers are asked to transcribe WWI personnel files from the Archives of New Zealand. Contained in these files are multiple types of documents (e.g. attestation papers, history sheets, death notifications), each with their own particular structure and fields. The Mark workflow begins by asking the user which type of document is being presented to them. Based on how they answer this initial prompt, a specific series of entities appropriate to the document type selected is presented to them for marking. This branching identification workflow would not have been possible in the 2012 iteration of Scribe because all documents were assumed to contain the same entities. Users are also able to make multiple marks, and to view existing marks made by other users.

Measuring the Anzacs: workflow begins with prompt to identify document type, determining which set of entities the user will be asked to mark

Workflow-wise, *Ensemble V.1* (like *Old Weather* before it) presented a single workflow in which contributors were asked to mark and transcribe all of the entities on a playbill in a single pass without viewing the annotations made by others. *Ensemble V.2* broke the consolidated mark/transcribe workflow into the three distinct workflows (Mark, Transcribe, and Validate) of today's Scribe. The aim in *Ensemble V.2*, as in the new Scribe framework, was to make participation less daunting by reducing the task burden to a single activity, considerably reducing the time and intellectual effort required of any one user.

Ensemble V.2 interface: Mark workflow

The other major change was to allow visitors to see the marks and transcriptions made by others. *Ensemble V.1* and *Old Weather* did not permit this in an effort to eliminate information cascade, a type of confirmation bias that causes people to base their actions on those of others, sometimes in contradiction of other data they possess. We decided this precaution was needlessly rigorous. By presenting others' marks and transcriptions — and allowing people to propose corrections — we thought we could potentially extract data that was “good enough” more quickly and with less duplicative effort.

Pause: NYPL Labs Restructuring
(May 2014 - October 2014)

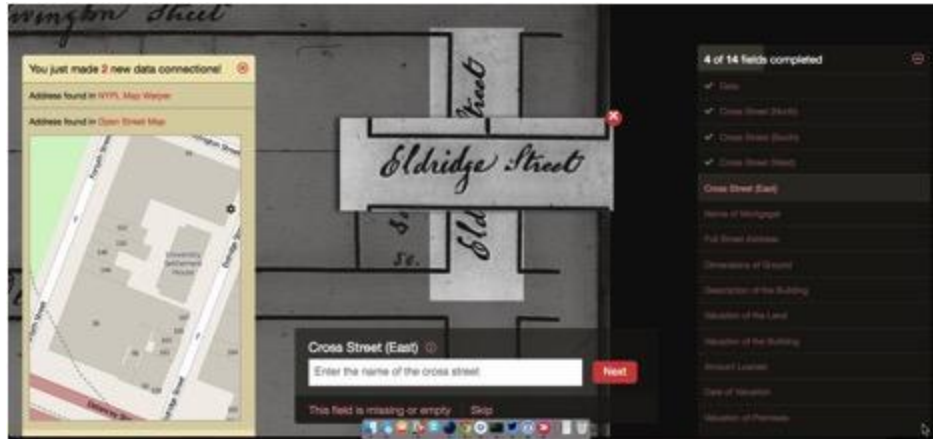
Work on Scribe was paused while NYPL Labs was restructured from an eight-person digital product group into a vertically integrated, 30-person digitization and library innovation program comprising digital imaging, metadata, permissions and reproductions, digital product development, and public programs and outreach. New organizational priorities stemming from a concurrent NYPL strategic planning process also redirected near-term focus onto rebuilding the Library's Digital Collections portal (<http://digitalcollections.nypl.org>). The main access point for free online access to NYPL's digitized visual and audiovisual content, Digital Collections features nearly a million digital objects including photographs, manuscripts, documents, moving images, audio recordings, maps, and other artifacts. Over time, this will become more closely integrated with the Library's crowdsourcing activities.

Sprint 2: Scribe API Sketches, Project Selection, and Interface Prototyping
(November 2014 - March 2015)

We refocused on Scribe beginning in November 2014, when Paul Beaudoin and David Riordan from the NYPL Labs team traveled to the Adler Planetarium in Chicago for an engineering meeting with Stuart Lynn and Sascha Ishikawa of Zooniverse. In the months that followed, Beaudoin and Lynn continued to sketch and document the Scribe API model while Brian Foo (NYPL) and Ishikawa focused on user interface prototypes. It was during this period that the three example projects that would eventually demonstrate Scribe in action — *Emigrant City*, *Old Weather: Whaling*, and *Measuring the Anzacs* — were formally selected. The third project, *Anzacs*, began development at Zooniverse's hub at the University of Minnesota and brought developer Andrea Simenstad into the core Scribe team.



Zooniverse interface prototype around ships' logs from New Bedford Whaling Museum



NYPL Labs interface prototype around *Emigrant Savings Bank* mortgage and bond ledgers

Sprint 3: Completing Codebase and Documentation; Project Launches
(April 2015 - November 2015)

In April 2014 the Scribe team regrouped in New York to plan the final push. Work involved testing and finalizing the code base, as well as its three constituent projects. The core technology stack supporting Scribe consists of a Ruby on Rails back-end speaking to a Mongo database, and a front-end consisting of HTML manipulated by Javascript. To this, we added the following specific technologies:

- Node, which simplifies Javascript package management, allowing us to depend on external, pre-written libraries for functionality that would be complicated and inefficient to write ourselves.
- Coffeescript, a scripting language that compiles to Javascript and simplifies implementing common patterns like class definitions, inheritance, list comprehensions, and closures. We find it to be a more natural and efficient language in which to script front-end functionality.
- CoffeReact.js, a relatively new framework that enforces modular component design and unidirectional inter-component communication. These constraints aided testing, allowed us to isolate work among team members, and provided a readymade standard against which to code, which was a great help when collaborating with remote programmers.

By the end of the summer, the Scribe code base was functionally complete and the development team progressively worked through major remaining issues. Test instances of all three example projects were deployed and underwent multiple rounds of user beta testing at NYPL. *Measuring the Anzacs* went live in early October, and *Emigrant City* launched a month later. *Old Weather: Whaling* is still in beta, but will launch in the coming weeks. In addition to publicity around each of these individual projects, there has also been an NYPL blog post about the Scribe project itself, “Scribe: Toward a General Framework for Community Transcription” (<http://www.nypl.org/blog/2015/11/23/scribe-framework-community-transcription>). This delves deeper into the Scribe framework, and serves as an introduction to its capabilities for historians, scientists, technologists, and others who might use it to power crowdsourced transcription projects.

IV. The Results So Far:

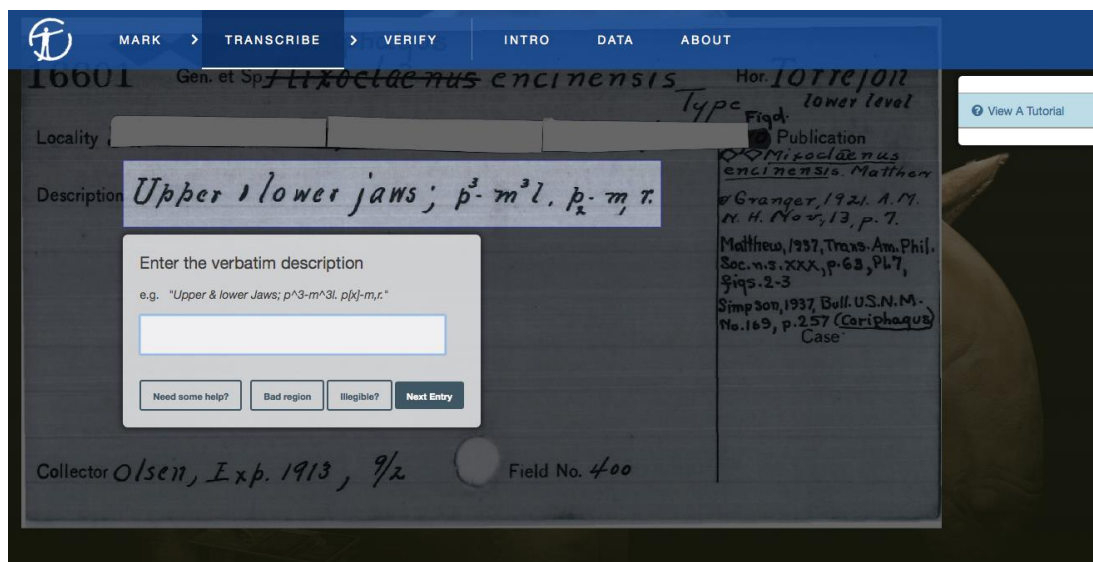
Although *Emigrant City* and *Measuring the Anzacs* have only recently launched, both projects have already resulted in tens of thousands of contributions. As of November 23, 2015 nearly 3,000 unique users have contributed to *Emigrant City*, submitting an average of 76 classifications each. These total 227,638 data points are the results of all three Scribe workflows including marks, transcriptions, and verifications. *Measuring the Anzacs* does not make use of the Scribe verification workflow, but as of the same date the project has elicited 93,826 marking classifications and 63,808 transcription classifications

from almost 7,000 unique users. More than 1,300 WWI enlistment documents for New Zealand soldiers have been fully processed and marked as complete. Currently, NYPL and Zooniverse plan to run these projects until all of their constituent documents have been fully processed.

NYPL has also had the opportunity to get first-hand feedback on the ease of using Scribe from developers unfamiliar with the codebase. On November 21-22, The American Museum of Natural History (AMNH) hosted their second annual hackathon. One team spent their 24 hours spinning up two proof-of-concept Scribe projects. The *Crowd Saurus* prototype was built around the records of 400,000 fossil mammal specimens in The AMNH's Childs Frick Building while the *Frick Fossil Finder* prototype was built around the shipping records of 200,000 of these specimens.

The team of five included two developers, none of whom had familiarity with the Scribe code base on either the technical end, or on the content end with organization of materials and workflows. They were able to go from zero prior knowledge to two complete, working prototypes. Setup included tasks such as creating and connecting a database, hosting the projects on a web server, configuration of the workflows, customization of the design, oauth integration, uploading a sample set of digitized materials, and tailoring help text to their content. During the process, the developers identified a few bugs which they documented in the Scribe Github repository. This is just the beginning of what we hope will be ongoing, collaborative work on Scribe by a growing network of contributors.

The strength of Scribe's configurability was evident in these prototypes' Mark Workflows in particular. At the start, both projects used as reference the Mark Workflow configuration of *Emigrant City*. However, the team decided to configure the *Crowd Saurus* marking stage to prompt each field successively, as opposed to all at once. This design decision was fully supported by the existing configurability of Scribe.

The image shows a web-based transcription interface for a document. The interface has a dark blue header with navigation tabs: MARK, TRANSCRIBE, VERIFY, INTRO, DATA, and ABOUT. The main content area displays a handwritten document with various fields for transcription. The 'Description' field is highlighted with a light blue box and contains the text 'Upper & lower jaws; p³-m³l. p₂-m₂r.'. Below this field is a text input box with the placeholder 'Enter the verbatim description' and an example 'e.g. "Upper & lower jaws; p^3-m^3l. p_2-m_2r."'. There are four buttons below the input box: 'Need some help?', 'Bad region', 'Illegible?', and 'Next Entry'. The 'Publication' field contains the text 'Matthew, 1937, Trans. Am. Phil. Soc. n.s. XXX, p. 63, Pl. 7, figs. 2-3'. The 'Collector' field contains the text 'Olsen, Exp. 1913, 9/2'. The 'Field No.' field contains the text '400'. A 'View A Tutorial' button is visible in the top right corner.

Crowd Saurus: transcription interface

Moving these prototypes to fully-functioning projects would require additional work including further attention to the help content and tutorials, uploading the full collection of digitized materials, and the sustained support of project leads to engage and support volunteers contributing to the projects. The hackathon being very recent, these remain as mere prototypes with no plans for further development. Even so, they represent an exciting and promising start for future uses of the Scribe code base, demonstrating that a viable crowdsourcing project can be established in less than two days.

V. The Future of Scribe:

The New York Public Library and Zooniverse will maintain the public code base in Github, and would also like to continue to improve it. Both organizations have identified further developments and changes that will help to make the product more usable. The current iteration of Scribe only encompasses three workflows, and so is only capable of capturing a limited range of data.

At NYPL, we believe Scribe shows strong potential as a repurposable framework for a wide range of crowdsourcing projects. A strategic goal for the coming two to three years is to bring disparate crowdsourcing projects together in an integrated "NYPL Public Works" platform, and to feed data produced by these projects into core collection records and public APIs via a linked open data layer. Scribe could be a key component of this larger platformization of crowdsourcing at NYPL.

One of the reasons there may not be more frameworks for community transcription of structured data is that it is difficult to design a single grammar to encode all of the decisions various workflows require. Scribe developed its own grammar organically through discussions between NYPL and Zooniverse, and by building out projects concurrently with building the underlying framework. Although the methods used by Scribe work well to support the needs of the current implementations, we would enjoy taking a step back at some point to carefully consider the inter-component and user-facing interfaces. We think the configuration options, for example, could be simplified by grouping related concerns and renaming a few things to more clearly express the system.

Related to the interfaces audit, we suspect Scribe would benefit from being more firmly split into front-end and back-end components. The framework is already fairly well divided, but formally defining the back-end API as distinct from the front-end components that read and write to it would make the system more modular, making it simpler to use only one or the other. Doing so would also simplify reasoning about and understanding the system.

An increasingly more pressing desire, however, is developing an interface to explore and vet the data assembled by the system. We spent a lot of time developing the parts that gather data, but perhaps not enough on interfaces to analyze it. Because we have reduced document transcription into several disconnected tasks, the process of reassembling the resulting data into a single, cohesive whole is complicated. That complexity requires a sophisticated interface to understand how a document's final set of assertions were derived from the chain of contributions that produced it. Fortunately, *Emigrant City* has produced ample contributions around which to build that interface, which we fully intend to do as the project continues.

Already, the Scribe release has prompted initial conversations with a range of potential adopters and collaborators representing a cross-section of key audience groups including digital humanities scholars, libraries, archives, and genealogy groups. As of this writing, the Scribe GitHub repository has attracted 32 watchers, has been starred 23 times, and forked five times — which compares favorably (especially considering the recency of the release) with the repository for Zooniverse's Panoptes API (<https://github.com/zooniverse/panoptes>), which supports user-created crowdsourcing projects (39 watchers, 38 stars, 17 forks).

The development and deployment of Scribe also has had a major effect on Zooniverse's strategy for meeting intense public demand for project creation. Scribe was designed to provide a tool for those who want to host their own projects, but who lacked the necessary development resources to run the full Zooniverse suite of software or to build their own project on top of it. It achieves this aim, giving Zooniverse a solution to offer for the many such projects which contact them each month. Furthermore, the process of developing workflow management which was necessary for Scribe informed the

subsequent implementation of a subject and workflow model in Panoptes, the most recent iteration of the software which supports hosted Zooniverse projects. Panoptes enables programmatic project creation for simple projects, a use case which requires the insights into project design that participating in Scribe development enabled. Furthermore, the projects which make use of Scribe, along with other text transcription efforts including partnerships with the Folger Library and with Tate, are continuing to add to our collective understanding of how to build successful projects. As such, Scribe-based experiments could enable a critical next step for Panoptes: integration of simple text transcription. NYPL Labs and Zooniverse have already discussed exploring next steps for co-development in 2016 and beyond.